

REPORT ZCLASS_AND_INTERFACES .

PARAMETERS p_land1 TYPE land1_gp.

** CREATE AN INTERFACE FIRST....*

```
INTERFACE LIF_DATA.  
  METHODS : GET_DATA IMPORTING CNTRY_KEY TYPE LAND1_GP,  
             DOWNLOAD_DATA,  
             DISP_DATA.  
ENDINTERFACE.
```

** CUSTOMER CLASS.....*

```
CLASS LCL_CUSTOMER DEFINITION.  
  PUBLIC SECTION.  
    INTERFACES: LIF_DATA.  
    ALIASES: GET_DATA FOR LIF_DATA~GET_DATA,  
              DISP_DATA FOR LIF_DATA~DISP_DATA,  
              DOWNLOAD_DATA FOR  
LIF_DATA~DOWNLOAD_DATA.  
  
  PROTECTED SECTION.  
    DATA: IT_KNA1 TYPE TABLE OF KNA1.  
    DATA WA_KNA1 TYPE KNA1.  
ENDCLASS.
```

** IMPLEMENTATION OF CLASS CUSTOMER*

CLASS LCL_CUSTOMER IMPLEMENTATION.

method GET_DATA.

```
Select * into table it_kna1 from kna1 where land1 =
cntry_key.
endmethod.
```

Method DISP_DATA.

IF IT_KNA1 IS NOT INITIAL.

LOOP AT it_kna1 into wa_kna1.

```
write :/ sy-tabix, wa_kna1-KUNNR, wa_kna1-name1,  
wa_kna1-land1.
```

ENDLOOP.

ELSE.

WRITE :/ 'NO DATA FOUND IN CUSRTOMER`S DISP DATA'
COLOR 7.

ENDIF.

endmethod.

METHOD DOWNLOAD_DATA.

CALL FUNCTION 'GUI_DOWNLOAD'

EXPORTING

```
FILENAME = 'D:/CLASSES/EXAMPLE.TXT'
```

TABLES

DATA_TAB = IT_KNA1.

ENDMETHOD.

ENDCLASS.

** declare SALES class*

CLASS LCL_SALES DEFINITION INHERITING FROM LCL_CUSTOMER.

PUBLIC SECTION.

Methods: GET_HEADER_DATA,
display_header_data,
DISP_DATA REDEFINITION.

PRIVATE SECTION.

TYPES:BEGIN OF ty_vbak,
kunnr type kunnr,
vbeln type vbeln_va,
erdat TYPE erdat,
netwr TYPE netwr_ak,
END OF ty_vbak .

data:

IT_VBAK TYPE TABLE OF ty_vbak ,
WA_VBAK type TY_VBAK .
ENDCLASS.

** Implementation of SALES class*

CLASS LCL_SALES IMPLEMENTATION.

```
method DISP_DATA.  
WRITE :/ 'sub class`s disp_data method'.  
DATA : lv_lines TYPE l, wa_kna1 TYPE kna1.  
uline.  
  LOOP AT it_kna1 into wa_kna1.  
    WRITE :/ sy-tabix , sy-vline, wa_kna1-kunnr COLOR 2, sy-vline,  
wa_kna1-name1 COLOR 3, sy-vline, wa_kna1-land1 COLOR 5, sy-  
vline.  
  ENDLOOP.  
  ULINE.  
endmethod.
```

```
method GET_HEADER_DATA.  
  Select kunnr vbeln erdat netwr FROM VBAK INTO TABLE  
IT_VBAK  
  FOR ALL ENTRIES IN IT_KNA1 WHERE KUNNR = IT_KNA1-  
KUNNR.  
endmethod.
```

```
method DISPLAY_HEADER_DATA.  
IF IT_VBAK IS NOT INITIAL .  
  ULINE.  
  LOOP AT IT_VBAK INTO WA_VBAK.  
    WRITE :/ WA_VBAK-KUNNR COLOR 2, SY-VLINE, WA_VBAK-  
VBELN COLOR 3, SY-VLINE, WA_VBAK-ERDAT COLOR 4, SY-  
VLINE, WA_VBAK-NETWR COLOR 7, SY-VLINE.  
  ENDLOOP.  
  ULINE.  
ELSE.  
  WRITE / 'SORRY .. NO DATA FOUND IN HEADER' COLOR 4.  
ENDIF.  
endmethod.
```

ENDCLASS.

** CREATE OBJECTS.....*

DATA O1 TYPE REF TO LCL_SALES.

START-OF-SELECTION.

CREATE OBJECT O1.

CALL METHOD O1->GET_DATA EXPORTING CNTRY_KEY =
P_LAND1.

*CALL METHOD O1->lif_data~DISP_DATA.

CALL METHOD O1->DISP_DATA. " both will call local one

CALL METHOD O1->DOWNLOAD_DATA.

CALL METHOD O1->GET_HEADER_DATA.

CALL METHOD O1->DISPLAY_HEADER_DATA